randintlist

Creating random integer number lists, with multiple numbers or not, sorted or not.

Version 0.1.1 – 24/09/2024

Cédric Pierquet
c pierquet – at – outlook . fr
https://github.com/cpierquet/randintlist

---

10 numbers, between 1 and 100, without repetition:
14,73,76,94,81,35,85,77,78,31
The 5th value is:
81

10 numbers, between 1 and 100, without multiples of 5:
84,72,78,18,12,87,51,16,23,71
The 9th value is:
23

15 numbers, between 1 and 20, with repetition:
15,11,11,16,19,14,2,8,6,12,16,14,4,18,9
The last value is:
9

6 sorted numbers, between 1 and 51, without repetition:
ascending : 9,12,21,26,43,49
descending : 51>45>42>15>10>6

The luarandom package do the same things, but with the obligation to compile with LuaLaTeX.

Contents

1 Loading, useful packages **3**

2 The Macros **3**

2.1 Global usage 3

2.2 Generate the list 3

2.3 Accessing elements 4

2.4 Version française 4

3 Example **5**

4 History **6**

5 The code **6**

1 Loading, useful packages

In order to load randintlist, simply use:

```
\usepackage{randintlist}
```

Loaded packages are ifthen, simplekv, listofitems, randomlist, xintexpr and xstring.

2 The Macros

2.1 Global usage

Package randintlist supports the creation of random integer number lists where a number will appear only once or multiple times. Generated lists can te used with listofitems.

All engines TEX are compatible with this package.

2.2 Generate the list

```
%generate list
\randintlist[keys]{\macro}
```

Available keys are:

- min: minimum value (default 1);
- max: maximum value (default 50);
- nb: number of values (default 6);
- sep: separator for the list (default ,);
- sort: sorting options, within no/asc/dec (default no);
- repeat: boolean to authorize repeating values (default false);
- exclude: list of excluded values (default empty);
- seed: random seed value according to used packages (default -).

```
%default values
\randintlist{\mylistA}\mylistA
```

30,31,14,3,50,9

```
%10 between 1 and 50, with ascending
\randintlist[sort=asc,min=1,max=50,nb=10]{\mylistB}\mylistB
```

1,8,10,19,31,38,42,45,48,49

```
%15 between 1 and 50, with ascending and repetitions allowed
\randintlist[sort=asc,min=1,max=50,nb=15,repeat]{\mylistC}\mylistC
```

2,3,3,12,12,19,25,30,32,33,33,42,45,50,50

```
%15 between 1 and 50, without multiples of 5
\randintlist[%
    sort=asc,min=1,max=50,nb=15,repeat,%
    exclude={5,10,15,20,25,30,35,40,45,50}]%
    {\mylistC}\mylistC
```

1,1,3,7,8,22,26,27,31,36,36,37,38,48,49

```
%list used with listofitems
\randintlist{\mylistD}\mylistD\par
\readlist*\mylistused{\mylistD}\showitems{\mylistused}\par
\mylistused[1]; \mylistused[-1]
```

21,46,39,2,18,43
21 46 39 2 18 43
21; 43

2.3 Accessing elements

```
%accessing item
\getitemfromrandintlist[separator]{\macro}{index}
```

```
%with default keys
\randintlist{\mylistE}raw list: \mylistE\par
items list:\par
\xintFor* #1 in {\xintSeq{1}{6}}\do{\getitemfromrandintlist{\mylistE}{#1}\par}
first element: \getitemfromrandintlist{\mylistE}{1}
```

raw list: 26,4,19,5,23,42
items list:
26
4
19
5
23
42
first element: 26

2.4 Version française

Voilà les commandes en version française, la syntaxe et les clés ne seront pas explicitées.

```
%obtenir la liste
\ListeRandint[Min=..,Max=..,Nb=..,Repet=..,Graine=..,Tri=..,Sep=..,Exclure=..]{\macro}

%extraire un élément
\ExtraireEltListeRandint[sep]{\macro}{position}
```

```
%liste
\ListeRandint[Min=5,Max=15,Nb=7,Repet,Tri=croiss,Sep={/}]{\maliste}\maliste\\
%élément
\ExtraireEltListeRandint[/]{\maliste}{4}
```

6/6/6/8/10/13/14
8

3 Example

The following example uses Ti*k*Z, and comes from `luarandom`'s documentation.

```
\begin{tikzpicture}[scale=0.75]
    \randintlist[min=1,max=100,nb=100]{\mylistsquare}
    \draw[thin,gray] (0,0) grid (10,10) ;
    \foreach \i in {1,...,100}{%
        \xdef\tmpnumber{\getitemfromrandintlist{\mylistsquare}{\i}}%
        \xdef\tmpnumberrow{\xinteval{\xintiiRem{\i-1}{10}}}%
        \xdef\tmpnumbercol{\xinteval{\xintiiQuo{\i-1}{10}}}%
        \draw ({0.5+\tmpnumbercol},{0.5+\tmpnumberrow}) node {\tmpnumber} ;
    }%
\end{tikzpicture}
```

7	51	92	24	72	25	55	63	78	35
71	83	59	84	37	27	38	77	36	22
28	20	48	85	98	45	89	11	1	9
57	74	50	14	82	75	8	88	65	95
15	43	18	33	29	93	39	6	30	31
58	4	56	3	90	53	70	42	99	17
46	80	67	96	64	60	100	97	94	34
5	41	52	81	73	54	47	26	32	16
87	79	23	62	13	12	68	91	86	69
40	76	19	44	10	66	2	61	49	21

4 History

0.1.1: Possibility to exclude values
0.1.0: Initial version

5 The code

```
% Author     : C. Pierquet
% licence    : Released under the LaTeX Project Public License v1.3c or later, see
     http://www.latex-project.org/lppl.txtf

\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{randintlist}[2024/09/24 0.1.1 Create a list of random numbers with or without multiple values]

%------History
% 0.1.1 Possibility to exclude values
% 0.1.0 Initial version

%------Packages
\RequirePackage{simplekv}
\RequirePackage{listofitems}
\RequirePackage{randomlist}
\RequirePackage{xintexpr}
\RequirePackage{xstring}
\RequirePackage{ifthen}

%-----Macros (latex3) for sorting and seed
\ExplSyntaxOn
\cs_new_eq:NN \randintseed \sys_gset_rand_seed:n
\NewDocumentCommand\intascsortlist{m}
{
    \clist_sort:Nn #1
    {
        \fp_compare:nNnTF {##1} > {##2}
        { \sort_return_swapped: }
        { \sort_return_same: }
    }
}
\NewDocumentCommand\intdessortlist{m}
{
    \clist_sort:Nn #1
    {
        \fp_compare:nNnTF {##1} < {##2}
        { \sort_return_swapped: }
        { \sort_return_same: }
    }
}
\ExplSyntaxOff

%----Internal macro (latex2) for testing if element is in list
\newcommand\ifintvalueinlist[2]{\IfSubStr{,#2,}{,#1,}}

\newcommand\boolvalueinlist[2]{\IfSubStr{,#2,}{,#1,}{\def\resisinlist{1}}{\def\resisinlist{0}}}

\newcommand\xintifintvalueinlist[4]{%
    \IfSubStr{,#2,}{,#1,}{\xdef\RESTMPVALUE{1}}{\xdef\RESTMPVALUE{0}}%
    \xintifboolexpr{ \RESTMPVALUE == 1}{#3}{#4}%
}

%----Macro for generating
\defKV[randomlistintegers]{%
    min=\def\TAEEmin{#1},%
    max=\def\TAEEmax{#1},%
    nb=\def\TAEEnb{#1},%
    sep=\def\TAEEsep{#1},%
    sort=\def\TAEEtri{#1},%
    seed=\def\TAEEseed{#1},%
    exclude=\def\TAEEexcluded{#1}
}
```

```
\setKVdefault[randomlistintegers]{%
    min=1,%
    max=50,%
    nb=6,%
    sep={,},%
    sort=no,%
    repeat=false,%
    seed={-},%
    exclude={}
}

\NewList{tmprandintlist}

\NewDocumentCommand\randintlist{ O{} m }{%1=keys,2=listname
    \useKVdefault[randomlistintegers]%
    \setKV[randomlistintegers]{#1}%
    \ifboolKV[randomlistintegers]{repeat}%repeat or not
        {%repeat allowed
            \IfStrEq{\TAEEseed}{-}%
                {}%
                {%
                    \randintseed{\TAEEseed}%
                }%
            %list creation of first element
            \def\resisinlist{1}%
            \whiledo{\resisinlist=1}{%
                \xdef\tmpresrandint{\fpeval{randint(\TAEEmin,\TAEEmax)}}%
                \boolvalueinlist{\tmpresrandint}{\TAEEexcluded}%
            }%
            \xdef#2{\tmpresrandint}%
            %list creation of other elements
            \xintFor* ##1 in {\xintSeq{2}{\TAEEnb}}%
                \do{%
                    \def\resisinlist{1}%
                    \whiledo{\resisinlist=1}{%
                        \xdef\tmpresrandint{\fpeval{randint(\TAEEmin,\TAEEmax)}}%
                        \boolvalueinlist{\tmpresrandint}{\TAEEexcluded}%
                    }%
                \xdef#2{#2,\tmpresrandint}%
                }%
        }%
        {%no repeating
            %randomize numbers
            \IfStrEq{\TAEEseed}{-}%
                {}%
                {%
                    \RLsetrandomseed{\TAEEseed}%
                }%
            \ClearList{tmprandintlist}%clearing the list
            \xintFor* ##1 in {\xintSeq{\TAEEmin}{\TAEEmax}}%
                \do{%
                    \ifintvalueinlist{##1}{\TAEEexcluded}%
                        {}%
                        {%
                            \InsertRandomItem{tmprandintlist}{##1}%
                        }%
                }%
            %list creation (first then other)
            \xdef#2{\tmprandintlist[0]}%
            \xintFor* ##1 in {\xintSeq{1}{\TAEEnb-1}}%
                \do{%
                    \xdef#2{#2,\tmprandintlist[##1]}%
                }%
        }%
    %sorting
    \IfStrEq{\TAEEtri}{asc}%if ascending
        {\intascsortlist{#2}}%
        {}%
    \IfStrEq{\TAEEtri}{des}%if descending
        {\intdessortlist{#2}}%
        {}%
```

```
    \StrSubstitute{#2}{,}{\TAEEsep}[#2]%swipping separator if necessary
}

%-----Macro for extracting
\NewDocumentCommand\getitemfromrandintlist{ O{,} m m }{%
    \IfEq{#1}{/}%
        {%
            \setsepchar[.]{#1}%
        }%
        {%
            \setsepchar{#1}%
        }%
    \readlist*\TMPLISTRANDINT{#2}%
    \TMPLISTRANDINT[#3]%
}
```